


[Subscribe \(Full Service\)](#) [Register \(Limited Service, Free\)](#) [Login](#)

 Search: ☐ The ACM Digital Library ☒ The Guide

THE GUIDE TO COMPUTING LITERATURE

[Feedback](#) [Report a problem](#) [Satisfaction survey](#)

 Terms used [inlining of virtual methods](#)

 Found **36,085** of **869,947**

Sort results by


[Save results to a Binder](#)
[Try an Advanced Search](#)
[Try this search in The Digital Library](#)

Display results


[Search Tips](#)
☐ Open results in a new window

Results 1 - 20 of 200

 Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

Best 200 shown

 Relevance scale ☐ ☐ ☐ ☐ ☐
1 [Reducing virtual call overheads in a Java VM just-in-time compiler](#)

Junpyo Lee, Byung-Sun Yang, Suhyun Kim, Kemal Ebcioglu, Erik Altman, Seungil Lee, Yoo C. Chung, Heungbok Lee, Je Hyung Lee, Soo-Mook Moon

 March 2000 **ACM SIGARCH Computer Architecture News**, Volume 28 Issue 1

 Full text available: [pdf\(994.66 KB\)](#) Additional Information: [full citation](#), [abstract](#), [index terms](#)

Java, an object-oriented language, uses *virtual methods* to support the extension and reuse of classes. Unfortunately, virtual method calls affect performance and thus require an efficient implementation, especially when just-in-time (JIT) compilation is done. *Inline caches* and *type feedback* are solutions used by compilers for dynamically-typed object-oriented languages such as SELF [1, 2, 3], where virtual call overheads are much more critical to performance than in Java. Wi ...

Keywords: Java JIT compilation, adaptive compilation, inline cache, type feedback, virtual method call

2 [Dynamic Adaptive compilation: Adaptive online context-sensitive inlining](#)

Kim Hazelwood, David Grove

 March 2003 **Proceedings of the international symposium on Code generation and optimization: feedback-directed and runtime optimization**

 Full text available: [pdf\(1.06 MB\)](#) [Publisher Site](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

As current trends in software development move toward more complex object-oriented programming, inlining has become a vital optimization that provides substantial performance improvements to C++ and Java programs. Yet, the aggressiveness of the inlining algorithm must be carefully monitored to effectively balance performance and code size. The state-of-the-art is to use profile information (associated with call edges) to guide inlining decisions. In the presence of virtual method calls, profile ...

3 [Practical virtual method call resolution for Java](#)

Vijay Sundaresan, Laurie Hendren, Chrislain Razafimahefa, Raja Vallée-Rai, Patrick Lam, Etienne Gagnon, Charles Godin

 October 2000 **ACM SIGPLAN Notices , Proceedings of the 15th ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications**, Volume 35 Issue 10

 Full text available: [pdf\(323.98 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)